



EXPERIMENT - 20

Object Oriented Programming Lab

Aim

Create a base class called SHAPE. Use this class to store two double type values. Derive two specific classes called TRIANGLE and RECTANGLE from the base class. Add to the base class, a member function getdata to initialize base class data members and another member function display to compute and display the area of figures. Make display a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes design a program that will accept driven of a TRIANGLE or RECTANGLE interactively and display the area.

Syeda Reeha Quasar

14114802719

4C7

EXPERIMENT – 20

Aim:

Create a base class called SHAPE. Use this class to store two double type values. Derive two specific classes called TRIANGLE and RECTANGLE from the base class. Add to the base class, a member function getdata to initialize base class data members and another member function display to compute and display the area of figures. Make display a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes design a program that will accept driven of a TRIANGLE or RECTANGLE interactively and display the area.

Source Code:

```
#include <iostream>

using namespace std;

class Shape
{
public:
    double height, base;

    Shape()
    {
        height = 0;
        base = 0;
    }

    void get_data()
    {
        cout << "\n\nEnter height/length and base/width to compute: ";
        cin >> height >> base;
    }

    virtual void display_area() {}
};

class Triangle : public Shape
```

```

{
public:
    void display_area()
    {
        cout << "\nArea of Triangle = " << (height * base) / 2;
    }
};

class Rectangle : public Shape
{
public:
    void display_area()
    {
        cout << "\nArea of Rectangle = " << height * base;
    }
};

int main()
{
    Shape *s;
    Triangle t;
    t.get_data();
    s = &t;
    s->display_area();
    Rectangle r;
    r.get_data();
    s = &r;
    s->display_area();
    return 0;
}

```

Output:

```
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ exp27.cpp -o exp27 } ; if ($?) { .\exp27 }
```

```
Enter height/length and base/width to compute: 12 8
```

```
Area of Triangle = 48
```

```
Enter height/length and base/width to compute: 23 54
```

```
Area of Rectangle = 1242
```

```
PS D:\sem 4\cpp\oops> █
```

```
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ exp27.cpp -o exp27 } ; if ($?) { .\exp27 }
```

```
Enter height/length and base/width to compute: 5 4
```

```
Area of Triangle = 10
```

```
Enter height/length and base/width to compute: 2 3
```

```
Area of Rectangle = 6
```

```
PS D:\sem 4\cpp\oops> █
```

Viva Questions

Q1) What is inheritance?

Ans.

Inheritance is one of the feature of Object Oriented Programming System(OOPs), it allows the child class to acquire the properties (the data members) and functionality (the member functions) of parent class.

Q2) What is a virtual function?

Ans.

A virtual function is a member function in the base class that we expect to redefine in derived classes.

Basically, a virtual function is used in the base class in order to ensure that the function is overridden. This especially applies to cases where a pointer of base class points to an object of a derived class.

Q3) What is child class?

Ans.

A class that inherits another class is known as child class, it is also known as derived class or subclass.

Q4) What is parent class?

Ans.

The class that is being inherited by other class is known as parent class, super class or base class.

Q5) What are the advantages of using inheritance in C++ Programming?

Ans.

The main advantages of inheritance are code reusability and readability. When child class inherits the properties and functionality of parent class, we need not to write the same code again in child class. This makes it easier to reuse the code, makes us write the less code and the code becomes much more readable.